



id: SI_004

model: anthropic/claude-sonnet-4.6

seeds:

04_risiko: mulighet

03_relasjon: tillit

05_handling: vent

01_perspektiv: bruker

02_tidhorisont: langsiktig

Instans SI_004

Seeds: bruker | langsiktig | tillit | mulighet | vent

Perspektiv: Tillit bygges sakte — muligheten ødelegges raskt

Den største blindsonen her er ikke teknisk. Det er at virksomheten planlegger å gi autonome agenter tillit de ikke har tjent, i systemer som ikke er forberedt på å gi den.

Vent. Ikke som passivitet, men som strategi.

Brukerens perspektiv avslører det tekniske problemet

Sluttbrukerne — saksbehandlere, innbyggere, interne fagpersoner — vil aldri se agentens “execution graph”. De ser bare konsekvensen: feil data, en sak som er endret uten årsak, en tjeneste som er utilgjengelig. Tillit til offentlige digitale tjenester tar 10 år å bygge og 10 minutter å rive ned. En runaway agent i et fagsystem med svak idempotens kan gjøre nettopp det.

Konkret risikomodell: Tre lag som svikter samtidig

1. **IAM-laget** mangler ephemeral credentials → agenten kjører med for brede, langlivede tokens → lateral movement-risiko er reel, ikke hypotetisk



2. **API-laget** har ingen kontraktsgarantier → én ikke-idempotent POST-kall som gjentas 3 ganger = 3 duplikate vedtak i fagsystemet
3. **Audit-laget** er distribuert → post-mortem er umulig → tillit kan ikke gjenopprettes fordi bevis mangler

Dette er ikke tre separate problemer. De er ett sammenvevd feilszenario.

Den langsiktige muligheten krever at du venter nå

Mainstream-rådet er: “start smått, iterer raskt.” Jeg argumenterer mot dette i offentlig sektor med legacy-landskap. Start **smalere og saktere**:

- Pilot kun på **read-only agenter** i 6 måneder — ingen skrivetilgang, ingen eksterne API-kall
- Bygg **én konsolidert audit-pipeline** (f.eks. OpenTelemetry + sentralisert SIEM) før agenten får handle
- Krev **OPA-evaluering som hard blokkering**, ikke advisory — ingen agent-handling uten policy-clearance

Neste steg

1. **Kartlegg alle ikke-idempotente endepunkter** agenten vil treffe — dette er din faktiske risikooverflate, ikke CVSS-scoren
2. **Definer “tillit-trappetrinn”**: agenten må bevise pålitelighet på read-only i 90 dager før skrivetilgang vurderes
3. **Sett bruker-observability som arkitekturkrav dag 1**: saksbehandleren skal alltid kunne se hvorfor agenten gjorde noe — ikke i logger, men i grensesnittet